App. No.: 10/010,674
Atty. Doc. No.: D02770

IN THE CLAIMS

Please amend the claims as follows:

1-3. (Currently cancelled)

4. (Previously amended) A method in a signal processor for quantizing a digital signal, the method comprising:

generating a fixed-point approximation of a value X÷D, wherein X is a fixed-point value based on one or more samples in the digital signal, and wherein D is a fixed-point quantization parameter;

generating a correction; and

modifying the approximation with the correction wherein the generating the correction includes multiplying X by DR, wherein DR is $((2^n + k*(D/2))/D) * (2^n$ modulo D), wherein k is a non-negative number.

5. (Original) The method of claim 4, wherein X is based on a DCT coefficient.

6. (Original) The method of claim 5, wherein X is based on an absolute value of the DCT coefficient.

7. (Previously amended) The method of claim 5, wherein X = X' + D>>1, wherein X' is a fixed-point value based on a Discrete Cosine Transform (DCT) coefficient wherein ">>" symbolizes a right shift, and wherein D is a quantization scale.

8. (Original) The method of claim 5, wherein X = X' + D2>>1, wherein X' is a fixed-point value based on a DCT coefficient, and wherein D2 is another quantization parameter.

9. (Original) The method of claim 5, wherein D = 2*Q, wherein D' is $2^{n-1}/Q$, wherein DR is $((2^n + k*(Q/2))/Q)*(2^{n-1} \% Q)$, and wherein Q is a quantization scale.

10. (Original) The method of claim 9, wherein X = X' + (3*Q+2)>>2, wherein X' is a fixed-point value based on a DCT coefficient.

11. (Previously amended) The method of claim 9, wherein X is a maximum of zero and the difference of X' and Q/2(X=max{0, X'-Q/2}), wherein X' is a fixed-point value based on a DCT coefficient.

12. (Original) The method of claim 4, wherein modifying the approximation with the correction includes adding the approximation with the correction.

13. (Original) The method of claim 12, wherein n is a word length, wherein the approximation includes a most significant word (MSW(approximation)) and a least significant word (LSW(approximation)), wherein the correction includes a most significant word (MSW(correction)), and wherein adding the approximation with the correction includes:

adding MSW(correction) with LSW(approximation) to produce a sum;

right-shifting the sum by n bits; and

adding the sum with MSW(approximation).

14. (Original) The method of claim 13, wherein the signal processor is a microprocessor having an instruction for calculating a function (A+B+1)>>1, and wherein the step of adding MSW(correction) with LSW(approximation) and the step of right-shifting the sum by n bits include:

calculating (MSW(correction) + LSW(approximation) + 1 >> 1) using the instruction; and

right-shifting (MSW(correction) + LSW(approximation) + 1 >> 1) by n-1 bits.

15. (Previously amended) The method of claim 14, wherein the microprocessor is an Intel™ microprocessor with MMX™ technology, and wherein the instruction is a Packed Average Word (pavgw) instruction.

16-18. (Currently cancelled)

19. (Previously amended) A method in a signal processor for quantizing a digital signal, the method comprising:

generating a fixed-point approximation of a value X÷D, wherein X is a fixed-point value based on one or more samples in the digital signal, and wherein D is a fixed-point quantization parameter;

generating a correction;

modifying the approximation with the correction; and

generating X, wherein X = 32 * ABS(X') + SGN(X')*(D>>1), wherein X' is a fixed-point value based on a DCT coefficient, and wherein D is a quantization step.

20. (Original) The method of claim 19, wherein generating X includes generating X'' = 16*ABS(X') + SGN(X')*(D>>2).

21. (Original) The method of claim 20, wherein n is a word length, and wherein generating the approximation includes:

multiplying X'' by D' to produce a most significant word of X''*D' (MSW(X''*D')) and a least significant word of X''*D' (LSW(X''*D')), wherein D' is $2^n/D$, wherein n is a positive integer such that $2^n > D$.

22. (Original) The method of claim 21, wherein generating the approximation further includes:

left-shifting MSW(X''*D') by one bit to produce MSW(X''*D')<<1;

right-shifting LSW(X''*D') by 15 bits to produce LSW(X''*D')>>15; and

bit-wise ORing MSW(X''*D')<<1 with LSW(X''*D')>>15.

23. (Original) The method of claim 21, wherein generating the correction includes:

multiplying X'' by DR to produce a most significant word of X''*DR (MSW(X''*DR)), wherein DR is $((2^n + k*(D/2))/D) * (2^n \% D)$, wherein k is a non-negative number.

24. (Original) The method of claim 23, wherein the step of adding the approximation with the correction includes:

left-shifting LSW(X''*D') by one bit to produce LSW(X''*D')<<1;

left-shifting MSW(X''*DR) by one bit to produce MSW(X''*DR)<<1;

adding LSW(X''*D')<<1 with MSW(X''*DR)<<1 to produce a sum;

right-shifting the sum by n bits; and

adding the sum with the bit-wise OR of MSW(X''*D')<<1 with LSW((X''*D')>>15.

25. (Original) The method of claim 24, further including, prior to the step of right-shifting the sum, adding D' to the sum if D>>1 is odd.

26. (Original) The method of claim 25, wherein the signal processor is a microprocessor having an instruction for calculating the function (A+B+1)>>1, and wherein the steps of adding LSW(X''*D')<<1 with MSW(X''*DR)<<1, adding D' to the sum, and right-shifting the sum by n bits include:

generating sum = (LSW(X''*D')<<1 + MSW(X''*DR)<<1 + 1) >> 1 using the instruction;

generating sum = (sum + (D'/2) +1) >> 1 using the instruction; and

right-shifting the sum by n-2 bits.

27. (Original) The method of claim 26, wherein the microprocessor is an Intel™ microprocessor with MMX™ technology, and wherein the instruction is the pavgw instruction.

28-33. (Currently cancelled)